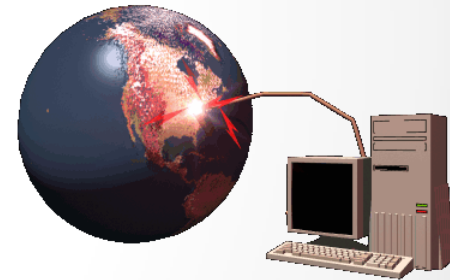




Teknik Informatika S1

Rekayasa Perangkat Lunak Lanjut

Web App. Process and Architecture



Disusun Oleh:

Egia Rosi Subhiyakto, M.Kom, M.CS

Teknik Informatika UDINUS

egia@dsn.dinus.ac.id

+6285640392988

SILABUS MATA KULIAH

8. Web App. Process and Architecture

9. WebE Design (1)

10. WebE Design (2)

11. Real Time Software

12. Testing Web App.

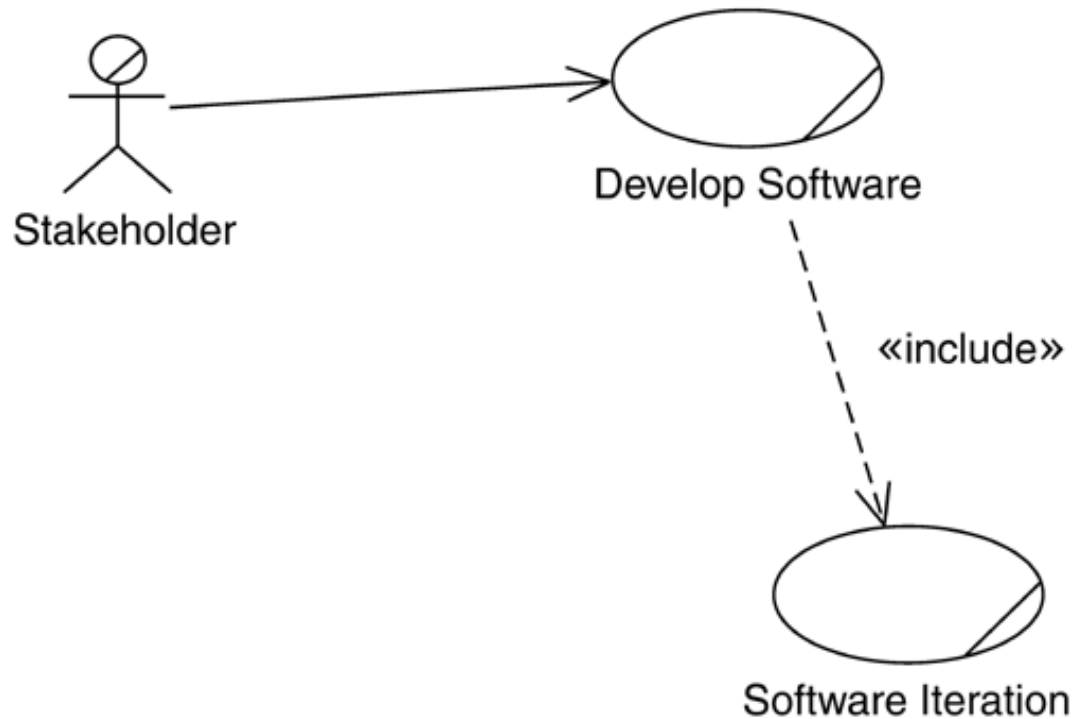
13. Present Tugas Besar

14. Present Tugas Besar

Agenda

- The process
 - Express the process in terms of UML model
- Defining the architecture
 - The activities
 - Web app architectural pattern
 - Presentation tier pattern

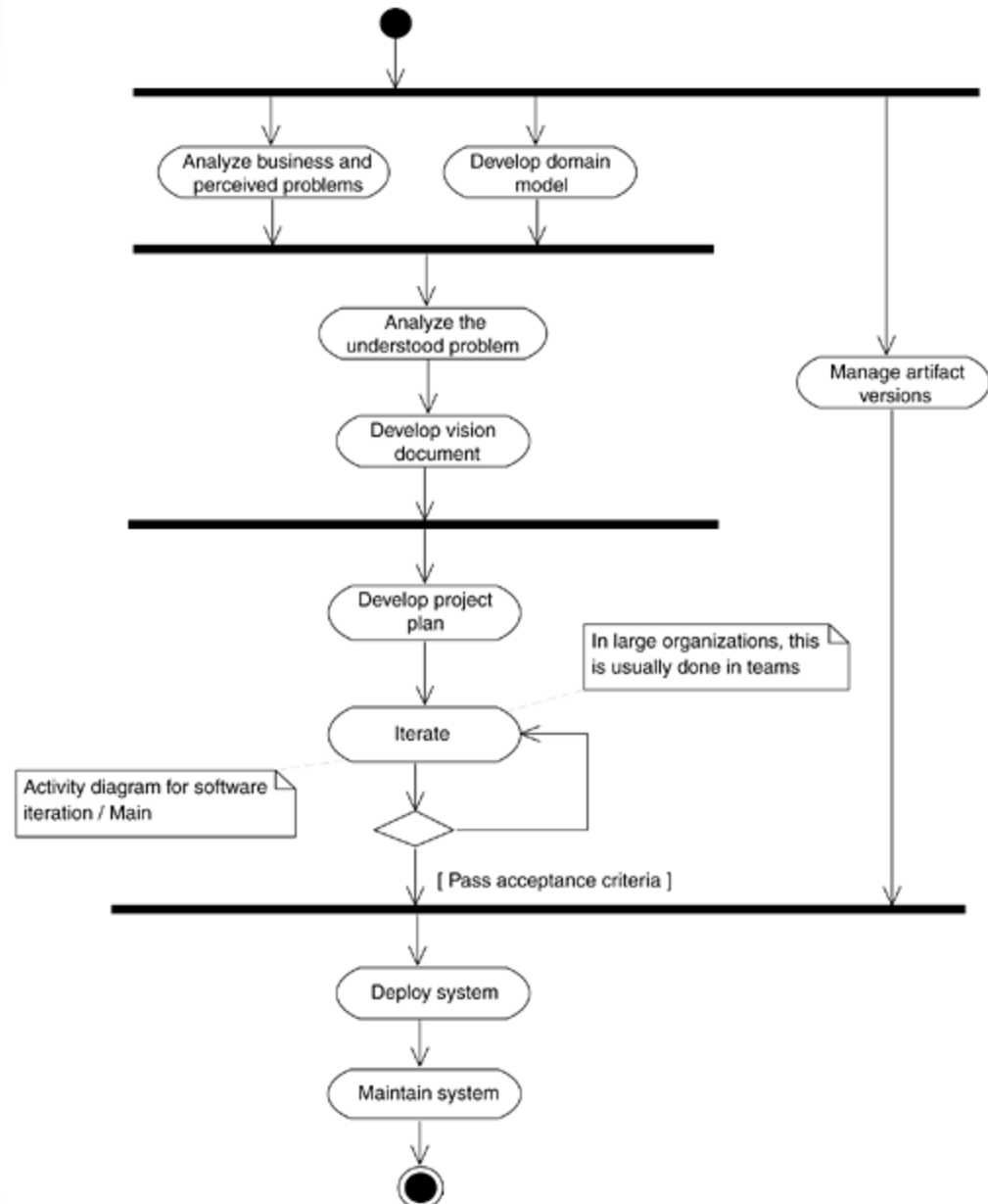
Pengembangan Software untuk Web App



Pengembangan Software untuk Web App (2)

- Pelaku usaha/ Stakeholder mewakili semua individu di luar proyek yang memiliki kepentingan dalam proyek.
 - Mereka tidak membangun perangkat lunak; mereka juga tidak bertanggung jawab atas artefak dari proses.
 - Mereka berkontribusi melalui interaksi dengan para pekerja dalam proses.
 - Stakeholders adalah pelanggan, eksekutif perusahaan, investor, dan pengguna: siapa pun dengan minat aktif dalam evolusi dan pengiriman sistem.
- Stakeholder berinteraksi dengan "bisnis" melalui business use case Develop Software, yang merupakan main use case dari proses.
- Sebuah bagian penting dari use case adalah meliputi use case Software Iteration.
 - Software Iterasi menjelaskan sebagian besar pekerjaan yang kita lakukan biasanya akan mempertimbangkan sebagai bagian dari proses pembangunan.

Activities of The Develop Software Use Case



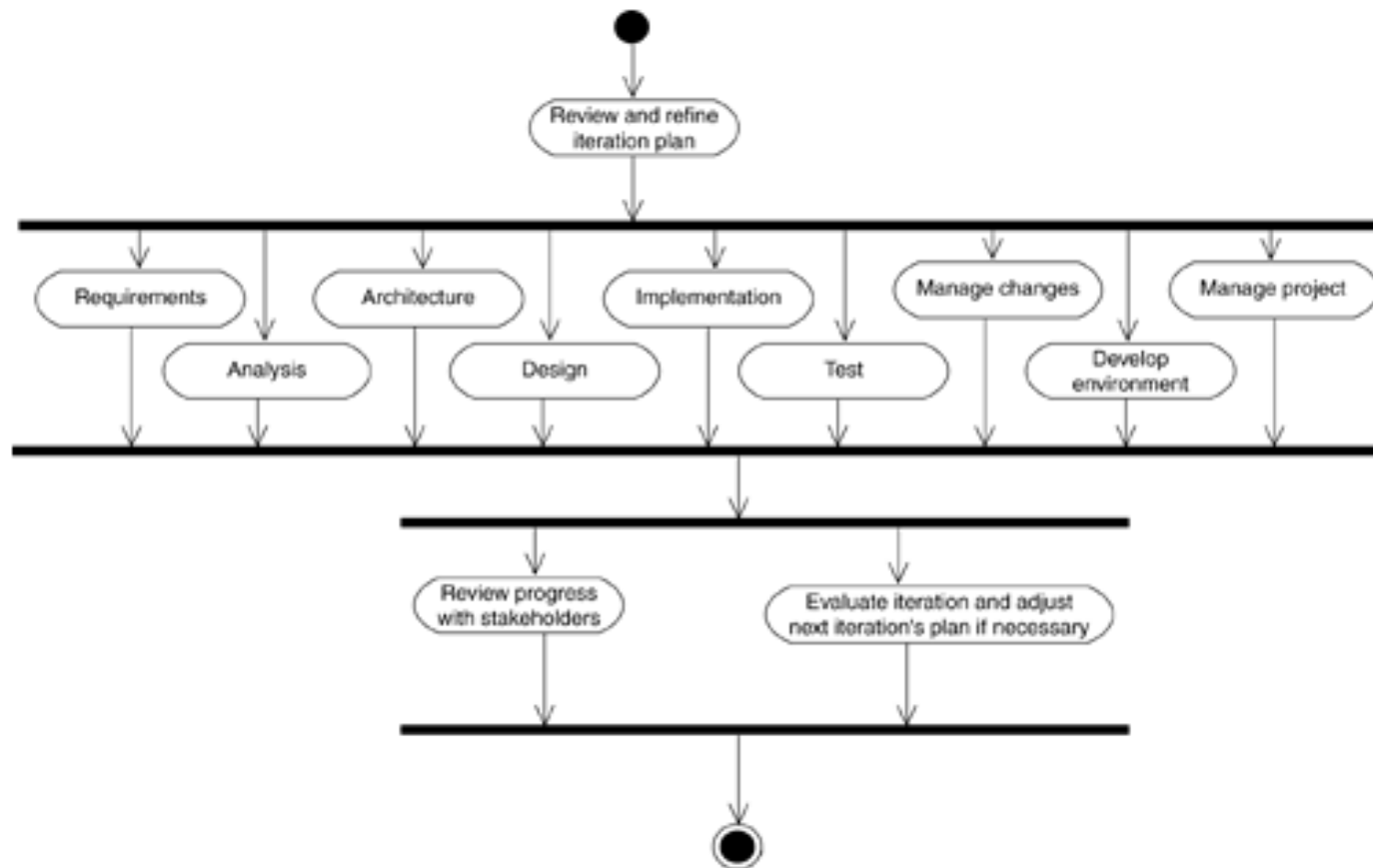
Activities of The Develop Software Use Case (2)

- **Mengelola versi artefak.**
 - Selesai bersamaan dengan setiap kegiatan lainnya
 - Ini adalah kegiatan proses manajemen perubahan dan penggunaan sistem kontrol versi.
- **Menganalisis bisnis dan masalah yang dirasakan.**
 - Mengambil objektif melihat keadaan bisnis.
 - Cobalah untuk tidak dipengaruhi oleh masalah yang disajikan oleh para pemangku kepentingan.
- **Mengembangkan model domain.**
 - Gunakan alat pengembangan perangkat lunak, seperti UML, persyaratan pengumpulan, dan aplikasi manajemen dokumen, untuk membangun sebuah model dari domain dan bisnis.
 - UML ekstensi untuk pemodelan bisnis adalah cara yang paling tepat untuk mengekspresikan entitas utama dan proses bisnis.
 - Artefak yang paling penting: daftar kata, yang mendefinisikan istilah kunci dan konsep dari konteks bahwa sistem bekerja.

Activities of The Develop Software Use Case (3)

- **Menganalisis masalah yang dipahami.**
 - Fokus pada masalah yang sebenarnya, atau setidaknya masalah bahwa para pemangku kepentingan bersedia untuk membiarkan Anda bekerja
- **Mengembangkan dokumen visi.**
 - Dokumen visi mengungkapkan ruang lingkup dan tujuan proyek perangkat lunak secara keseluruhan.
- **Mengembangkan rencana proyek.**
 - Rencana proyek menguraikan kegiatan usaha pengembangan seluruh, perangkat lunak mendefinisikan tonggak utama dan referensi dokumen standar yang tepat, termasuk perubahan dan manajemen konfigurasi rencana.
- **Menyebarkan sistem.**
 - Menyampaikan dan menginstal sistem.
- **Memelihara sistem.**
 - Pemeliharaan sistem pada dasarnya adalah mini version dari proses yang berkembang itu.

Activities of The Software Iteration Use Case



Activities of The Software Iteration Use Case (2)

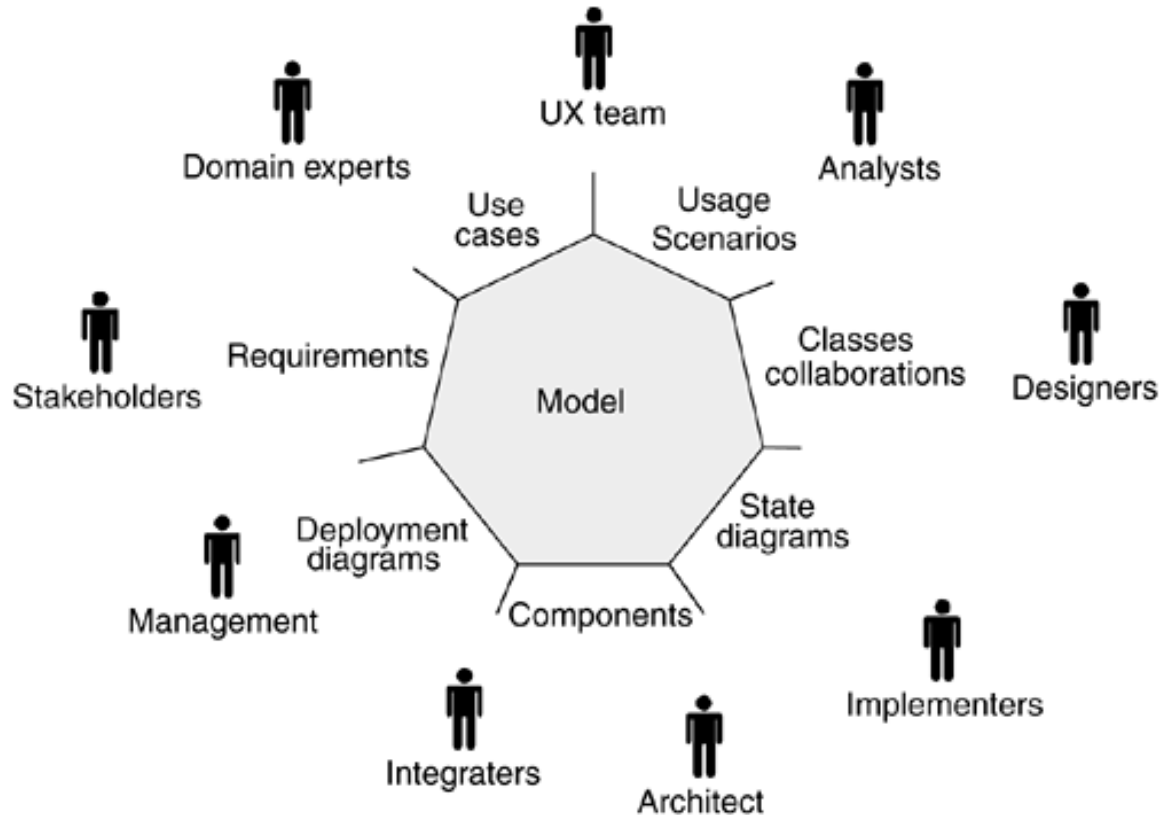
- **Meninjau dan menyempurnakan rencana iterasi.**
 - Sebelum iterasi, tim mengambil melihat rencana iterasi.
 - Manajer proyek membuat alokasi sumber daya, dan tim membuat komitmen penyampaian-nya.
- **Meninjau kemajuan dengan para pemangku kepentingan.**
 - Seringkali para pemangku kepentingan sangat tertarik pada kemajuan proyek.
- **Mengevaluasi iterasi dan menyesuaikan rencana iterasi berikutnya.**
 - Apakah kiriman telah dibuat dengan upaya yang diharapkan?
 - Apakah tingkat kualitas yang diharapkan oleh tim?
 - Apakah itu memenuhi harapan pribadi Anda?
 - Ini semua adalah langkah-langkah penting yang harus tercermin dalam rencana iterasi berikutnya.

Artifact of The Development Process

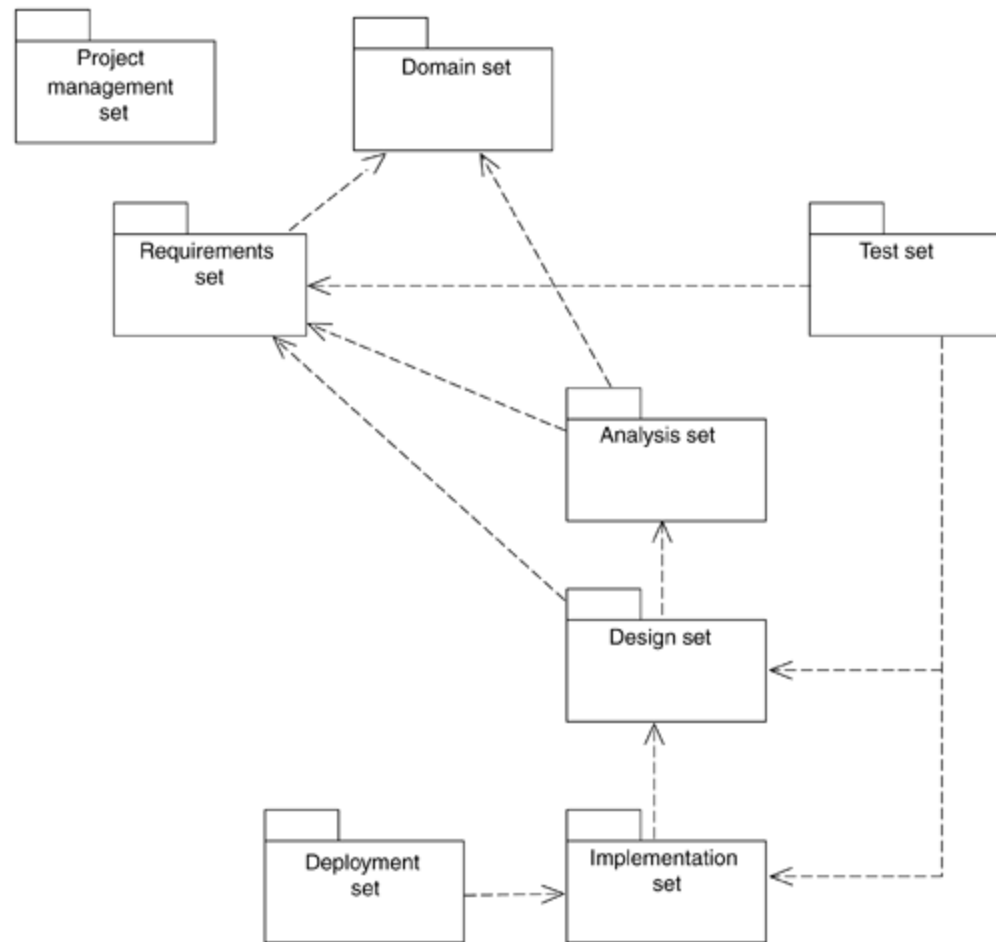
- Dalam proses pembangunan Aplikasi Web, model yang direkomendasikan adalah:
 - Domain model
 - Use case model
 - Analysis/design model
 - Implementation model
 - Process model
 - Security model
 - User experience, or user interface, model

Artifact of The Development Process

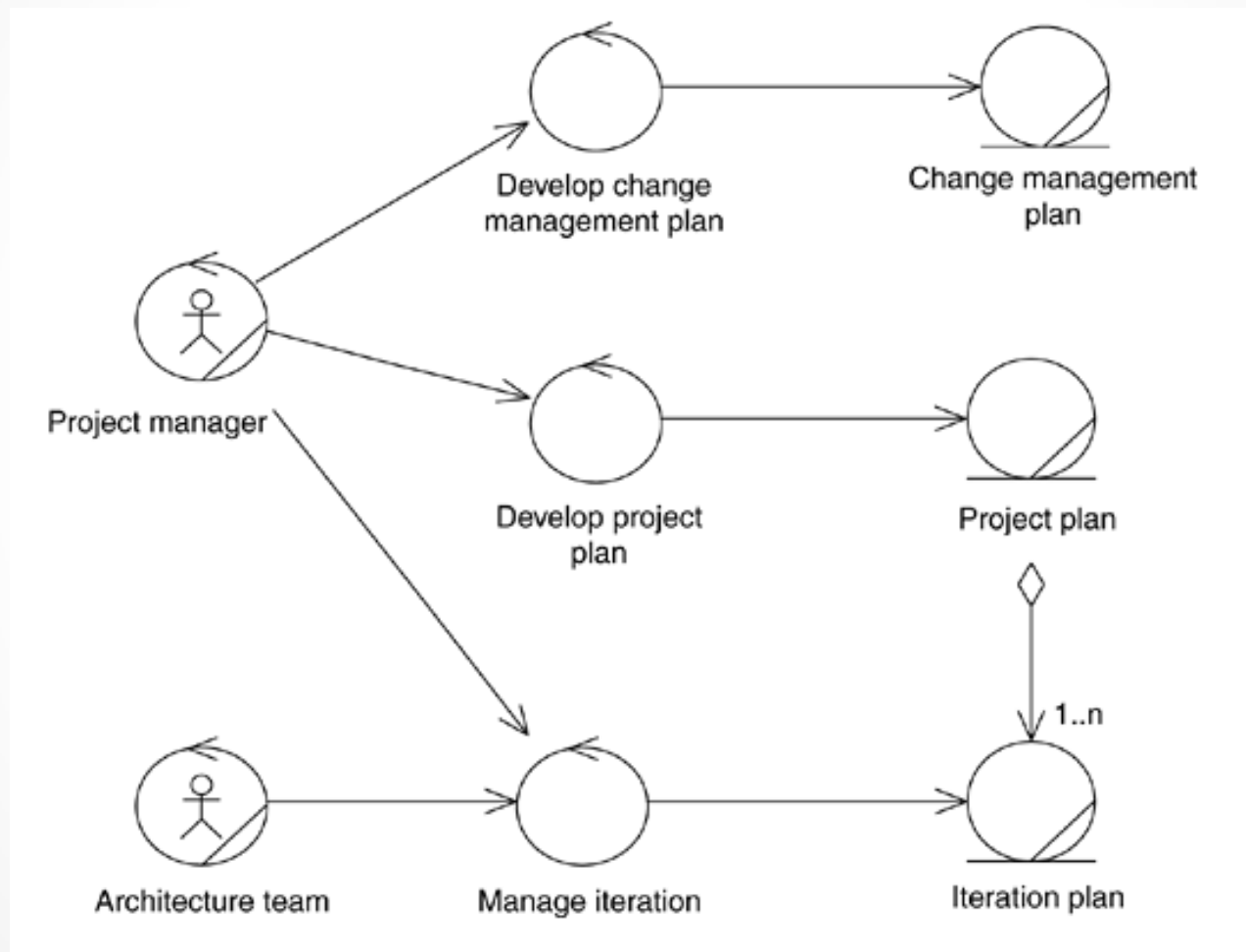
(2)



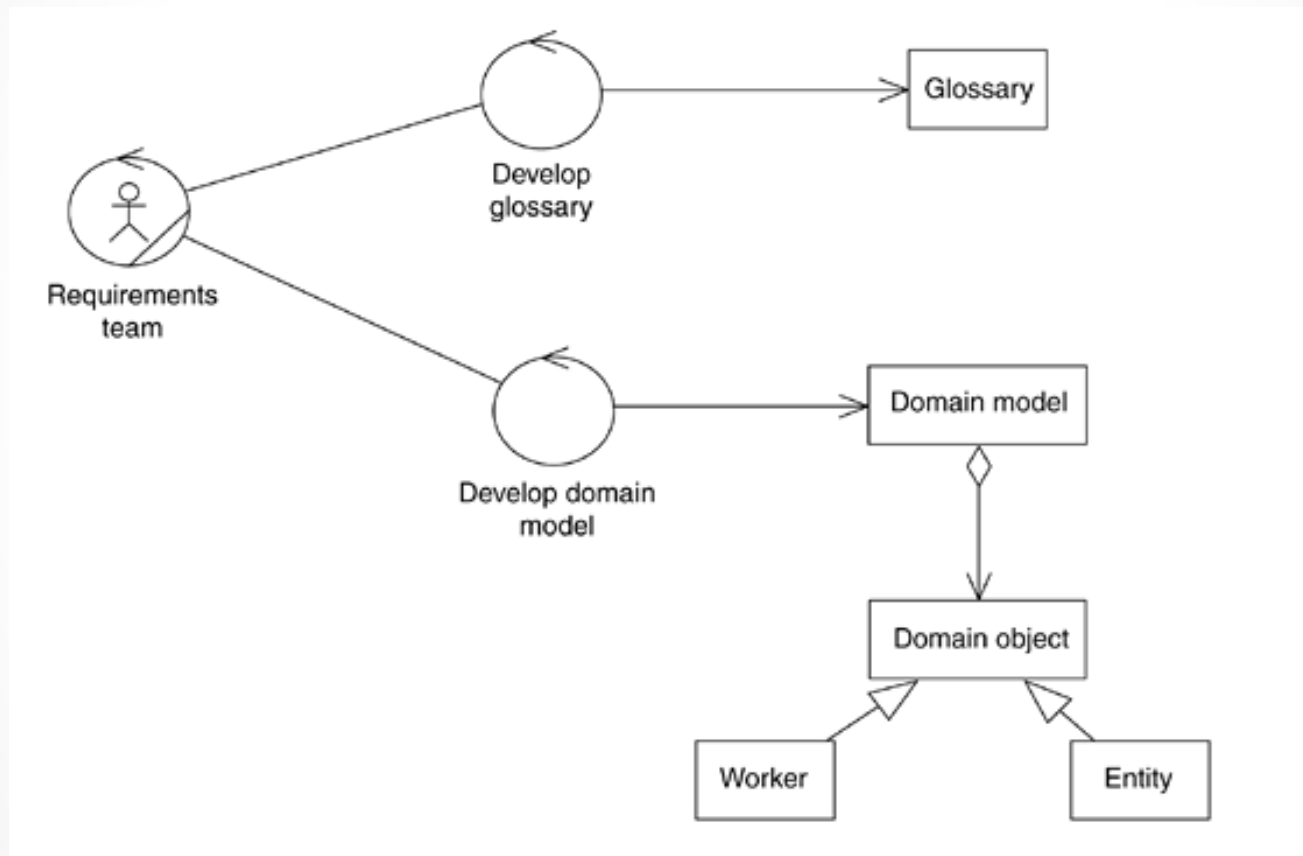
Dependencies and traceabilities in artifact sets



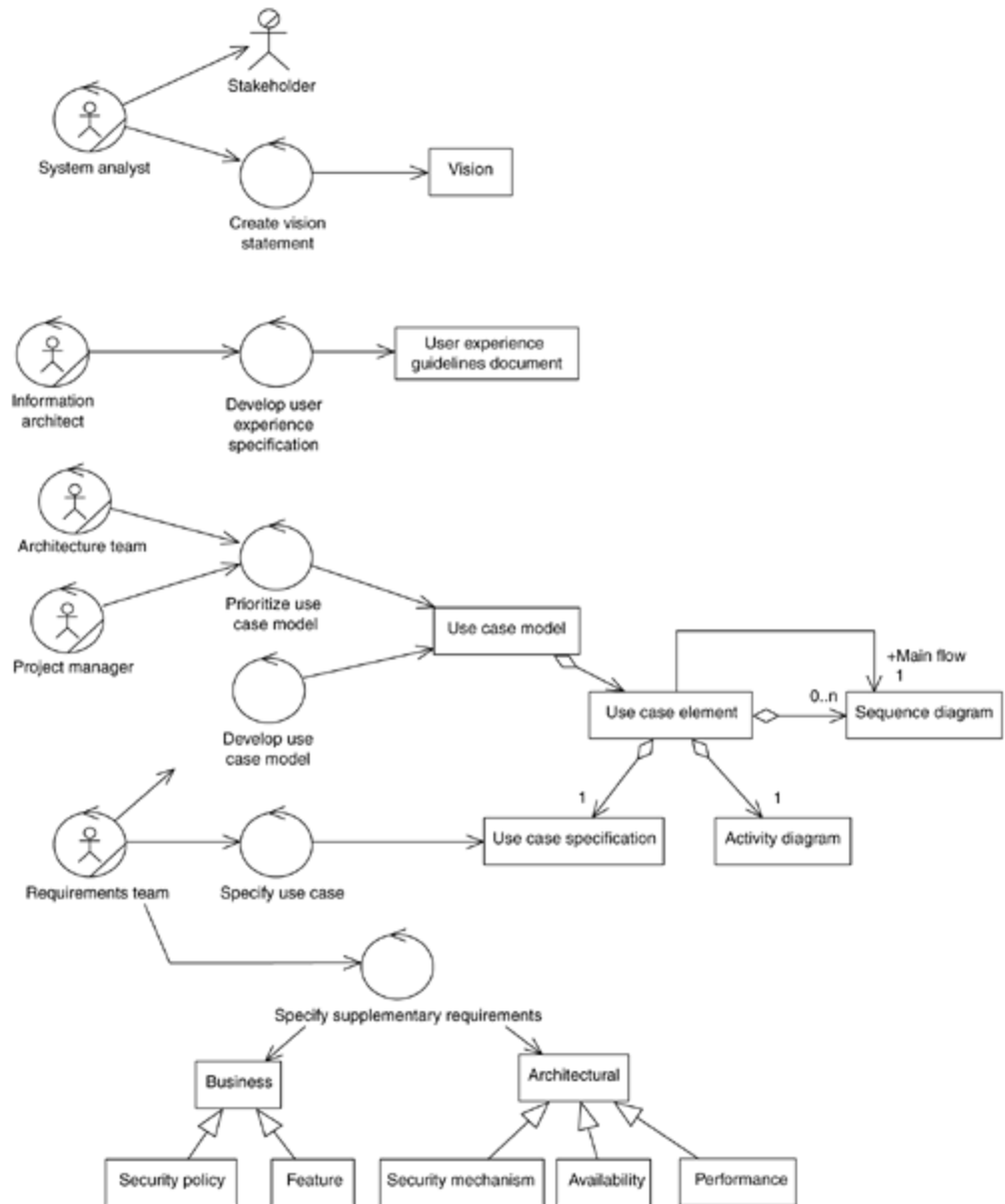
Project Management Set



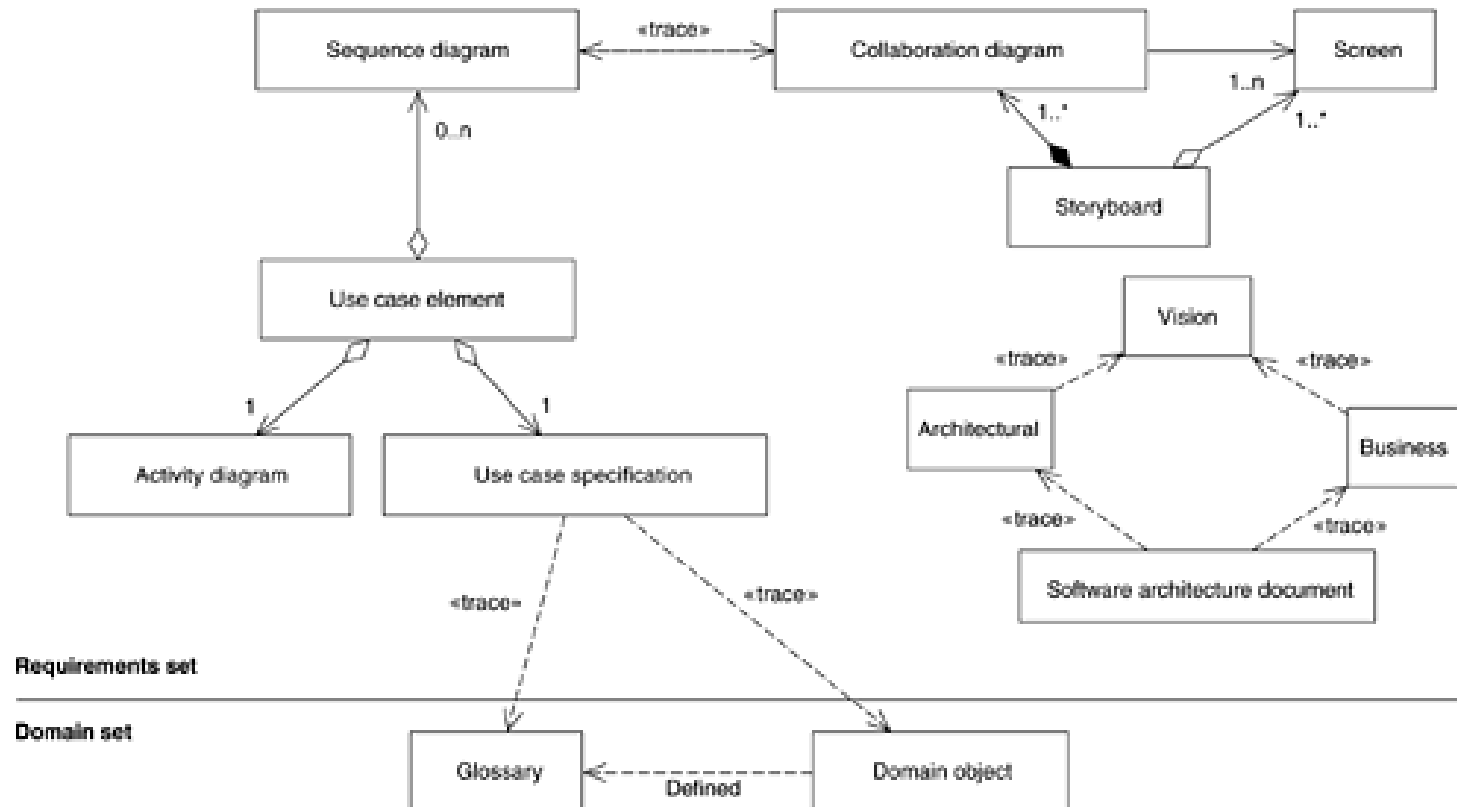
Domain set



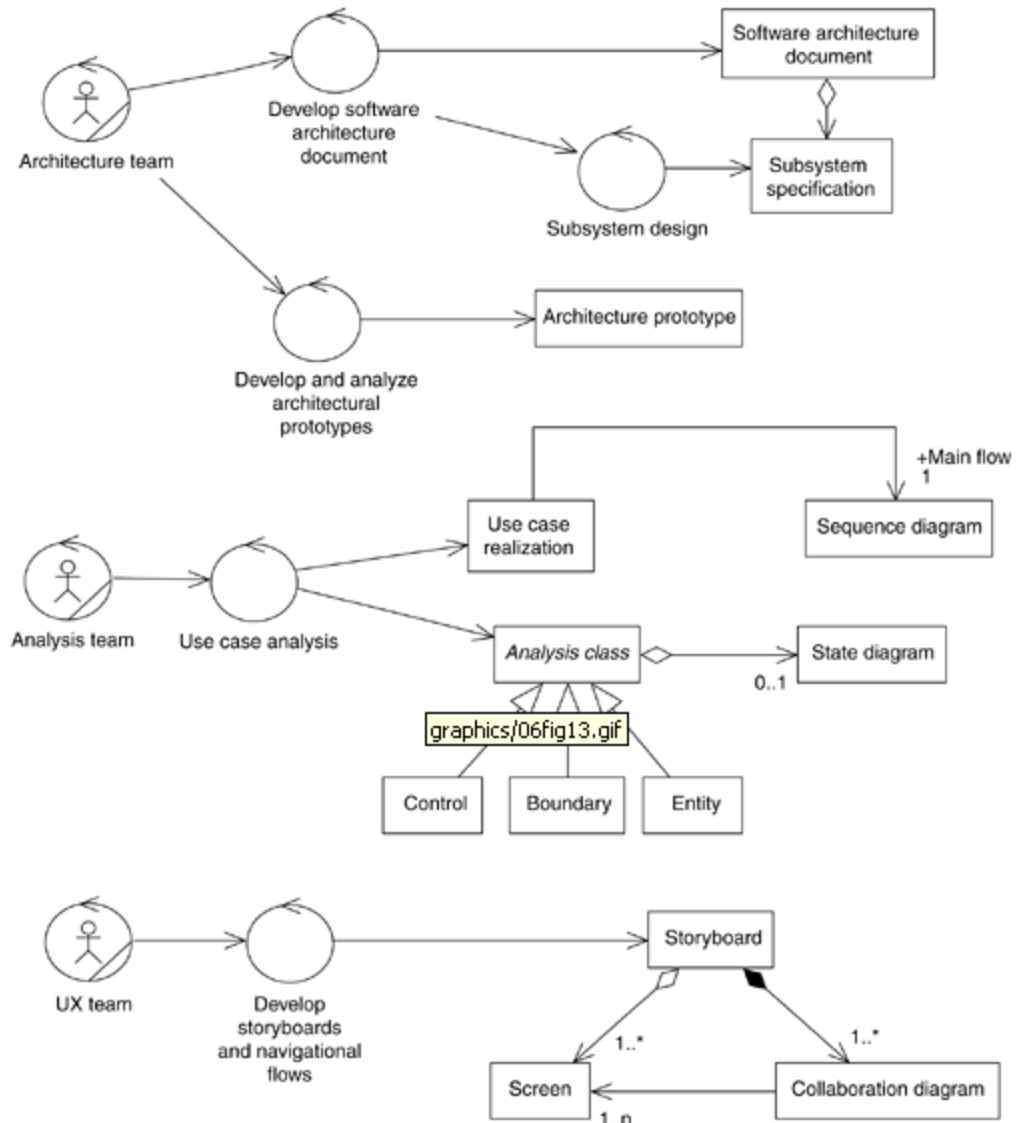
Requirements set



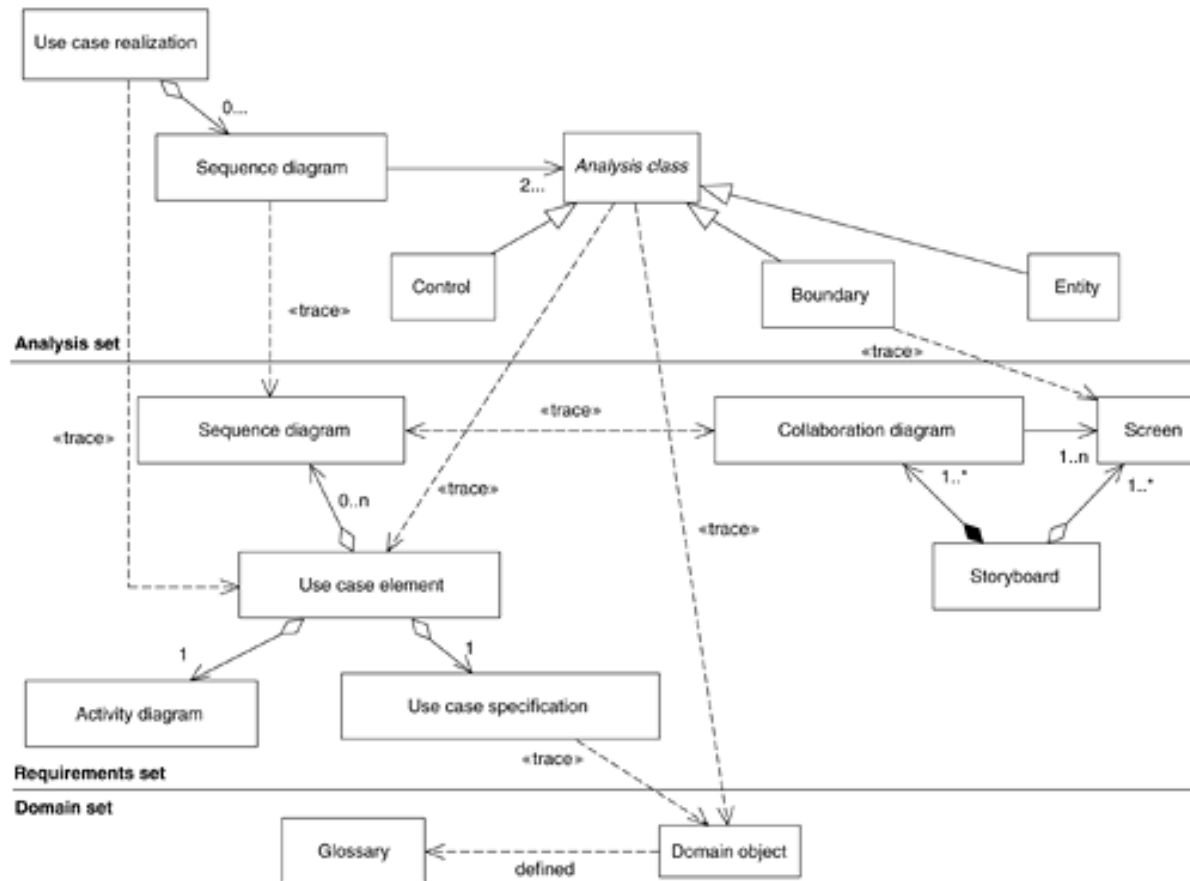
Associations and traceabilities in the requirements set



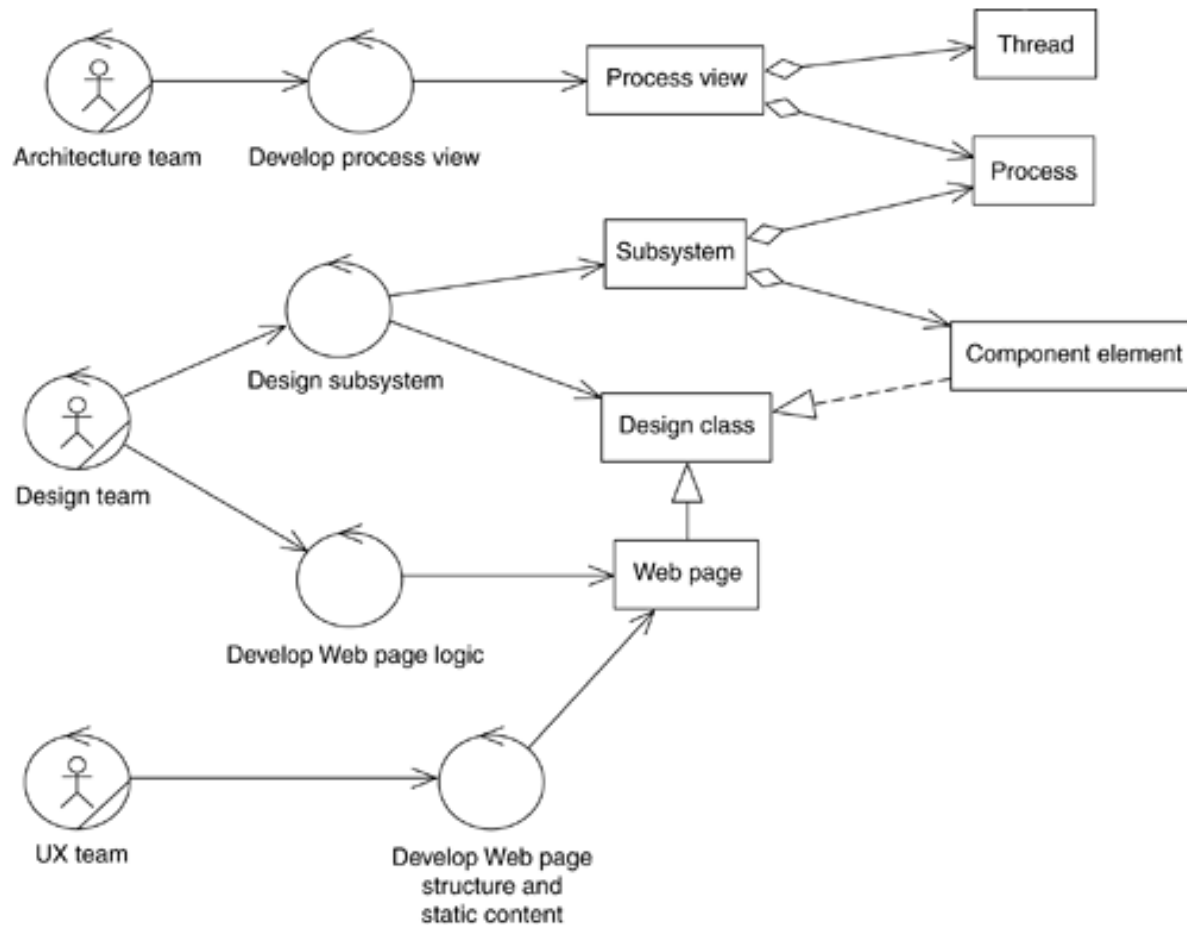
Analysis set



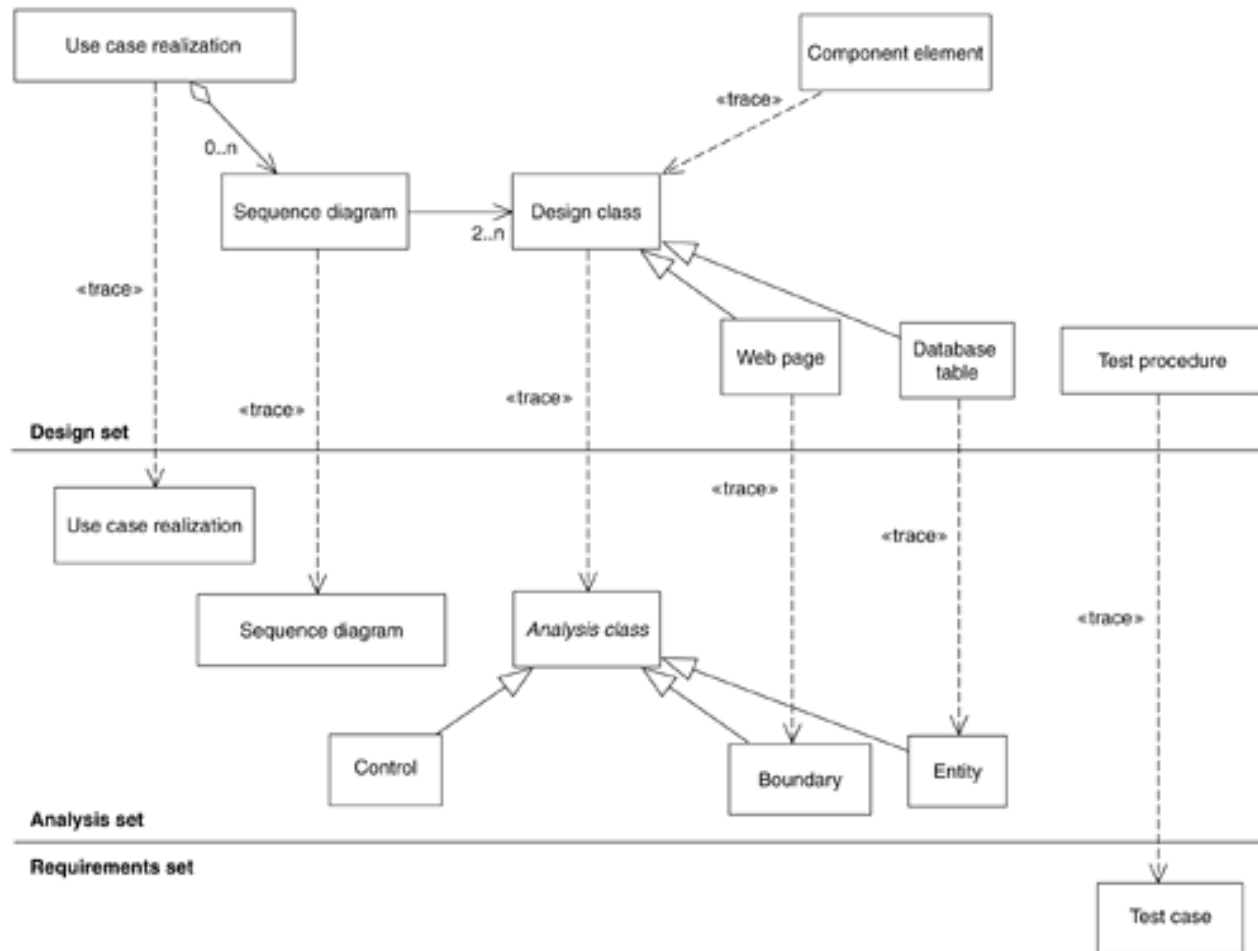
Associations and traceabilities in the analysis set



Design set

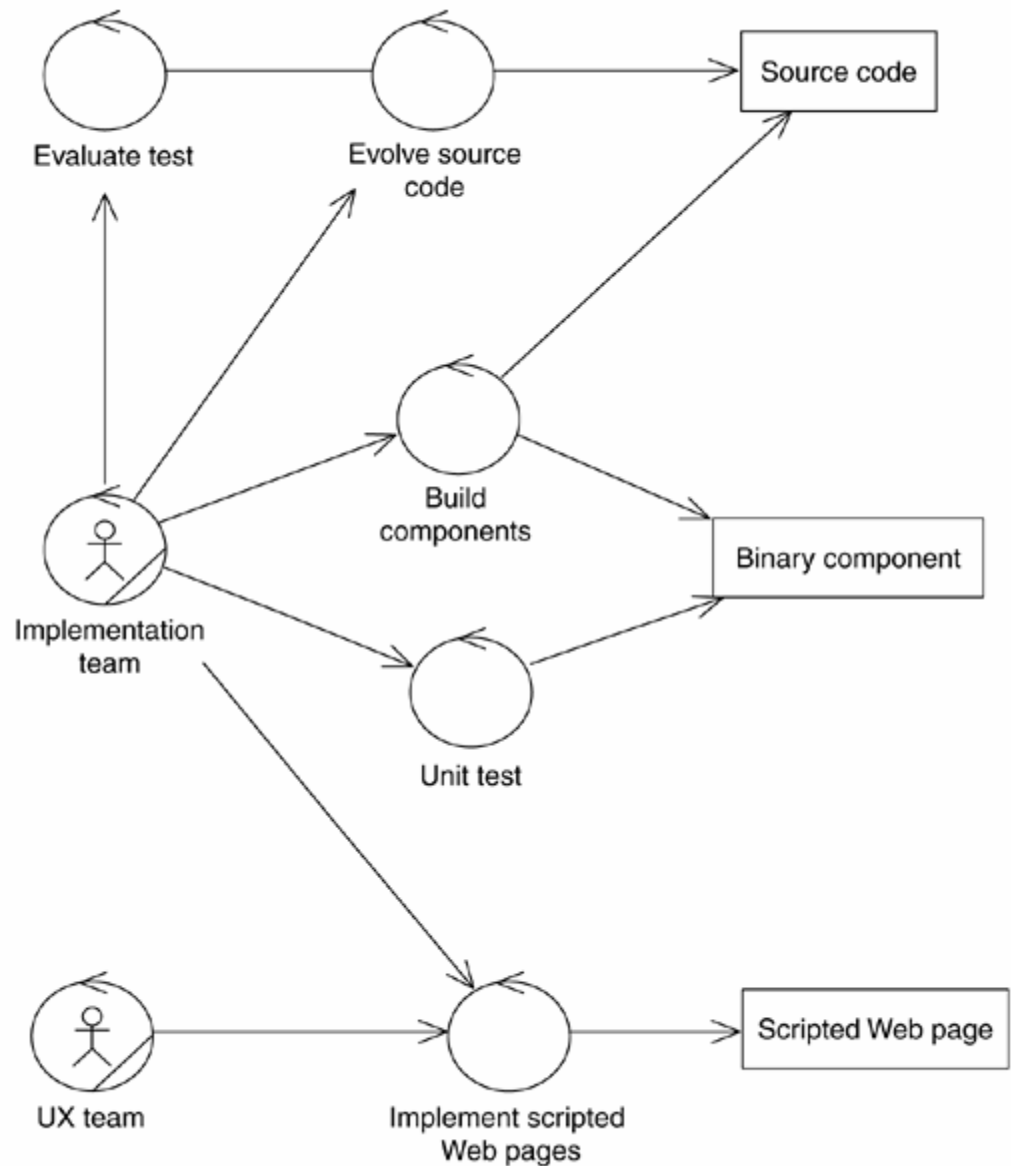


Associations and traceabilities in the design set

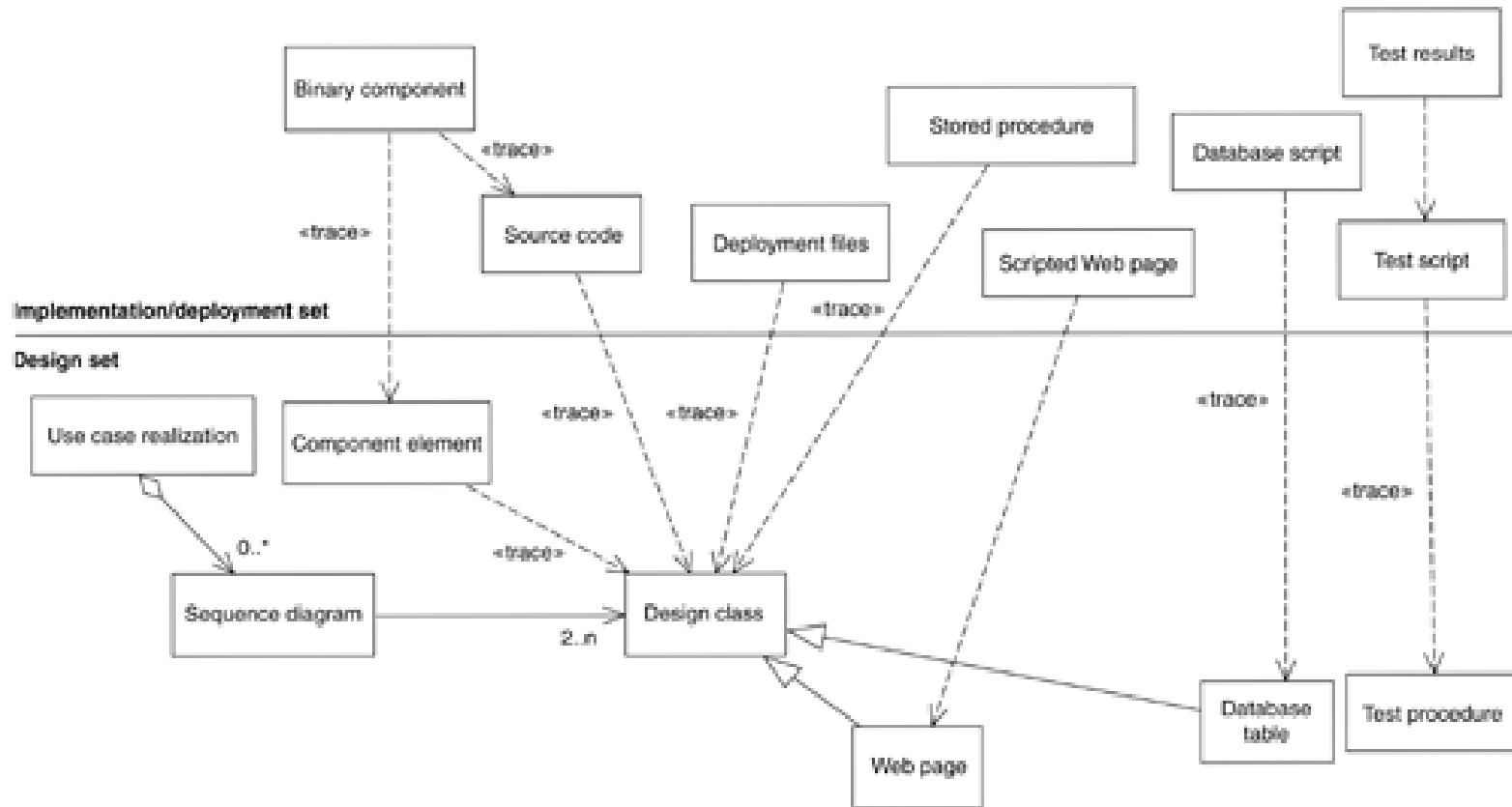


Implementation

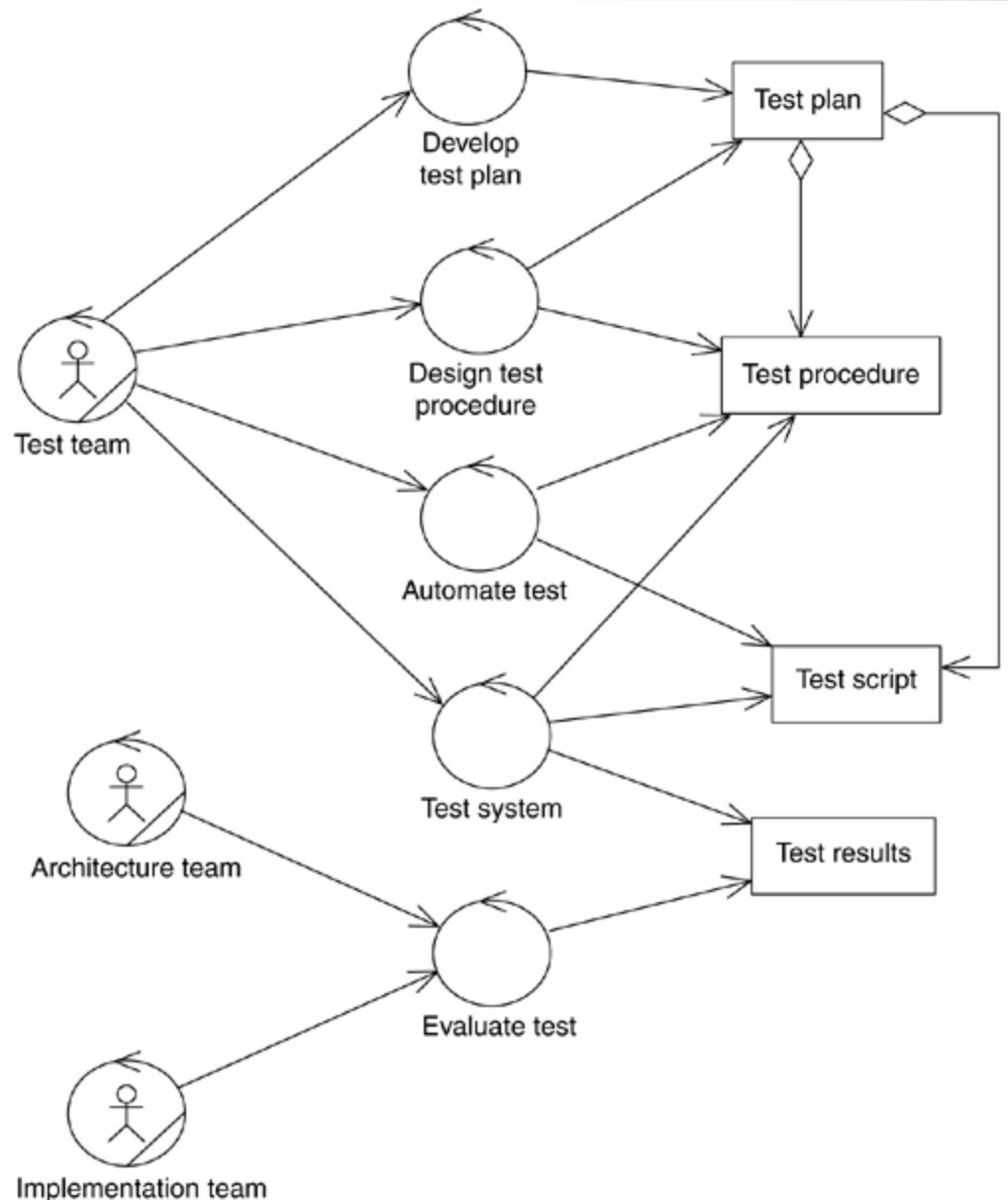
set



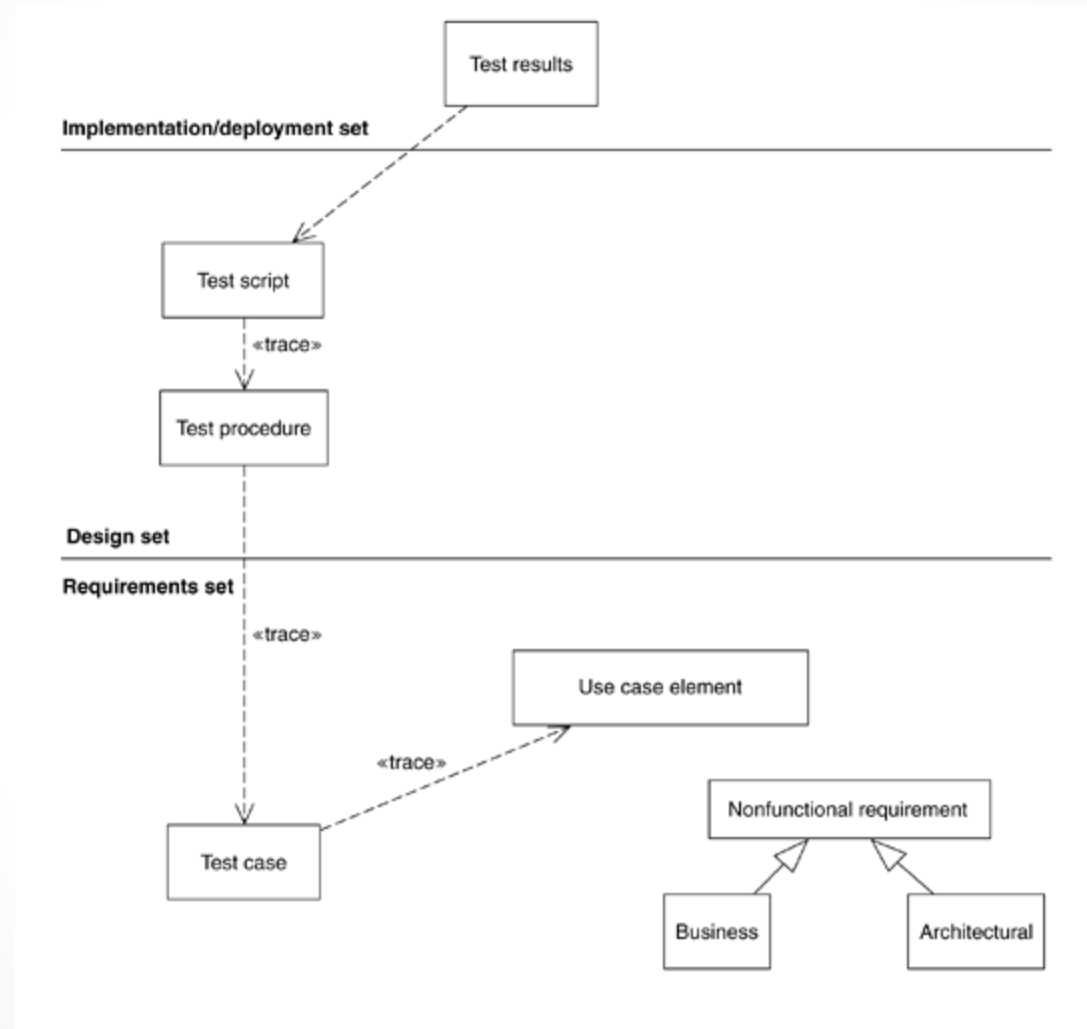
Associations and traceabilities in the implementation set



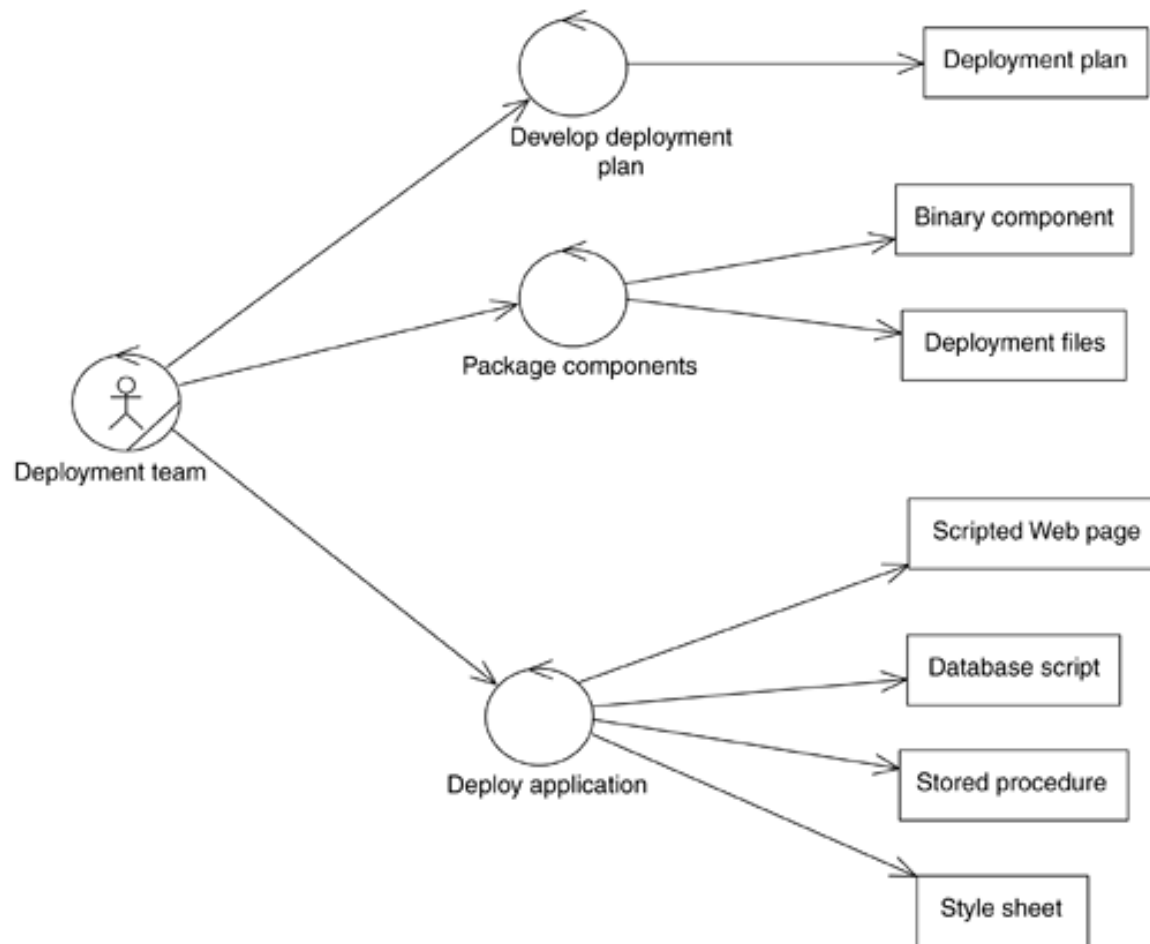
Test set



Associations and traceabilities in the test set



Deployment set



Definisi Architecture

- Kegiatan Arsitektur
- Pola arsitektur Web app:
 - Façade
 - Page Composition
 - Templated Page
- Presentation tier pattern:
 - Thin Web Client
 - Thick Web Client
 - Web Delivery

Kegiatan Architecture

- Memeriksa dan memprioritaskan Use Case dan mencari persyaratan arsitektur yang signifikan
- Menetapkan dan mendokumentasikan kandidat arsitektur dan mempersiapkan dan mengevaluasi prototipe.
- Tentukan strategi reuse (penggunaan kembali)

Pola Arsitektur Web Application

- **Façade**

- Informasi dinamis dalam setiap halaman Web yang diberikan mungkin harus dibangun dari koleksi obyek bisnis dan pengontrol.
- Kelas façade berpasangan dengan halaman Web dinamis. Setiap halaman web memiliki kelas façade khusus dirancang yang bertindak untuk mengkonsolidasikan semua objek orkestrasi bisnis dan untuk memberikan yang jelas, mudah digunakan antarmuka bagi penulis halaman Web script untuk menggunakan.

- **Page composition.**

- Setiap halaman Web konseptual dalam sistem ini berkumpul di runtime dari satu set independen fragmen halaman yang lebih kecil, yang sering digunakan kembali di halaman dalam sistem.
- Sebagai contoh, banyak aplikasi internet ritel menyediakan cara cepat untuk memasukkan kriteria pencarian produk pada setiap halaman Web konseptual.

- **Templated page.**

- Pola ini mendefinisikan satu halaman template yang semua halaman Web keluar melalui perjalanan mereka ke klien.
- Mirip dengan pola komposisi halaman, pola halaman templated menyediakan struktur tambahan template yang didefinisikan secara formal dan layar (halaman konseptual).

Patterns for The Presentation Tier

- **Web thin client pattern** digunakan terutama untuk aplikasi berbasis Internet, di mana ada sedikit kontrol dari konfigurasi klien.
 - Klien hanya membutuhkan bentuk standar browser Web.
 - Semua logika bisnis dijalankan di server.
- **Thick Web client pattern** digunakan ketika jumlah arsitektur signifikan dari logika bisnis dijalankan pada mesin klien.
 - Biasanya, klien menggunakan HTML dinamis, applet Java, atau kontrol ActiveX untuk menjalankan logika bisnis.
- **Web delivery pattern** digunakan ketika browser Web bertindak terutama sebagai pengiriman dan perangkat wadah untuk sistem objek terdistribusi.
 - Selain HTTP untuk klien dan server komunikasi, protokol lainnya, seperti IIOP dan DCOM, dapat digunakan untuk mendukung sistem objek terdistribusi.

Thin Web Client

Sudut Pandang Persyaratan

Persyaratan utama: kemampuan untuk memberikan sistem ke banyak jenis sistem pengguna mungkin

Aplikasi Thin Web client hanya membutuhkan kemampuan minimal pada klien

Persyaratan pembatas tambahan: kemampuan browser untuk menerima kembali cookie; kemampuan untuk beroperasi di lingkungan bandwidth rendah; dll

Thin Web Client (2)

Sudut Pandang Perancangan

- Komponen utama ada di server
- Komponen utama adalah sebagai berikut:
 - **Client browser:**
 - Setiap standar bentuk berkemampuan HTML browser.
 - **Web server:**
 - Jalur akses utama untuk semua browser klien.
 - **HTTP connection:**
 - Protokol yang paling umum digunakan antara browser klien dan Web Server.
 - **Static page:**
 - Sebuah halaman Web dengan UI dan informasi konten yang tidak pergi melalui pemrosesan di sisi server.
 - **Dynamic page:**
 - Web yang masuk melalui beberapa bentuk pemrosesan di sisi server

Thin Web Client (2)

- **Application server:**

- Mesin utama untuk mengeksekusi logika bisnis server-side.

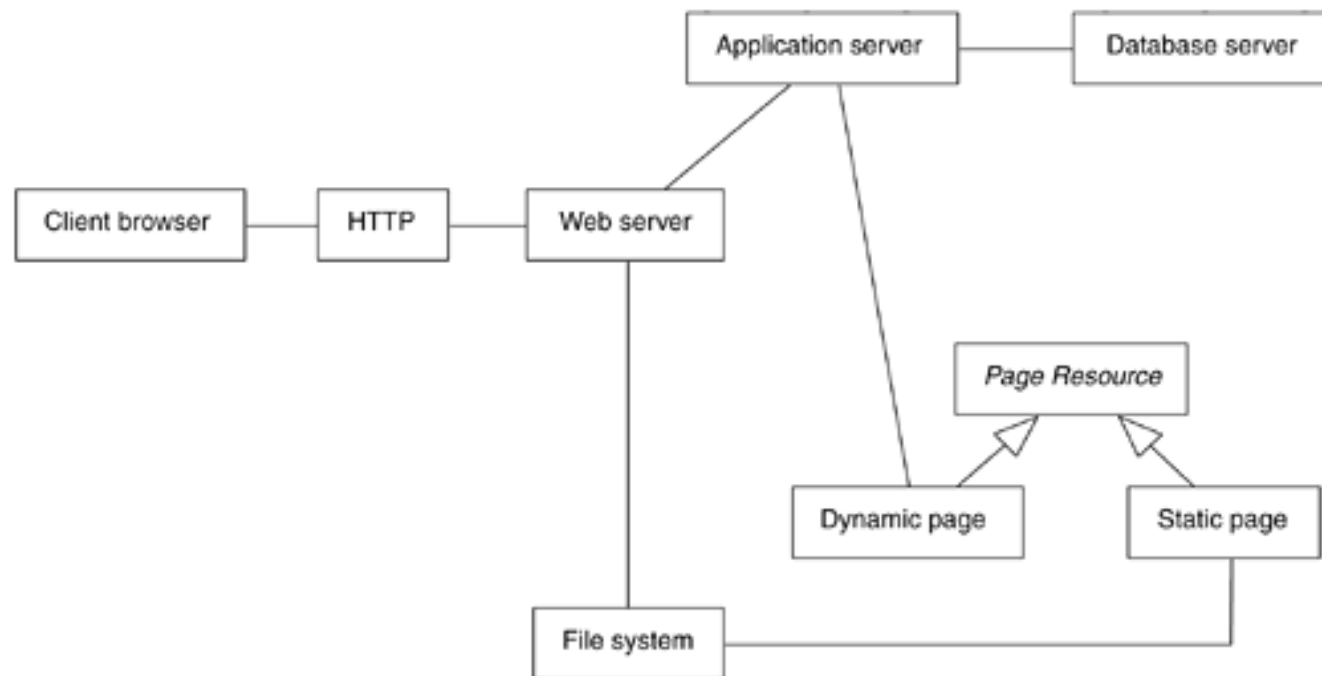
- **Database server:**

- Bagian dari sistem yang mempertahankan state terus-menerus.

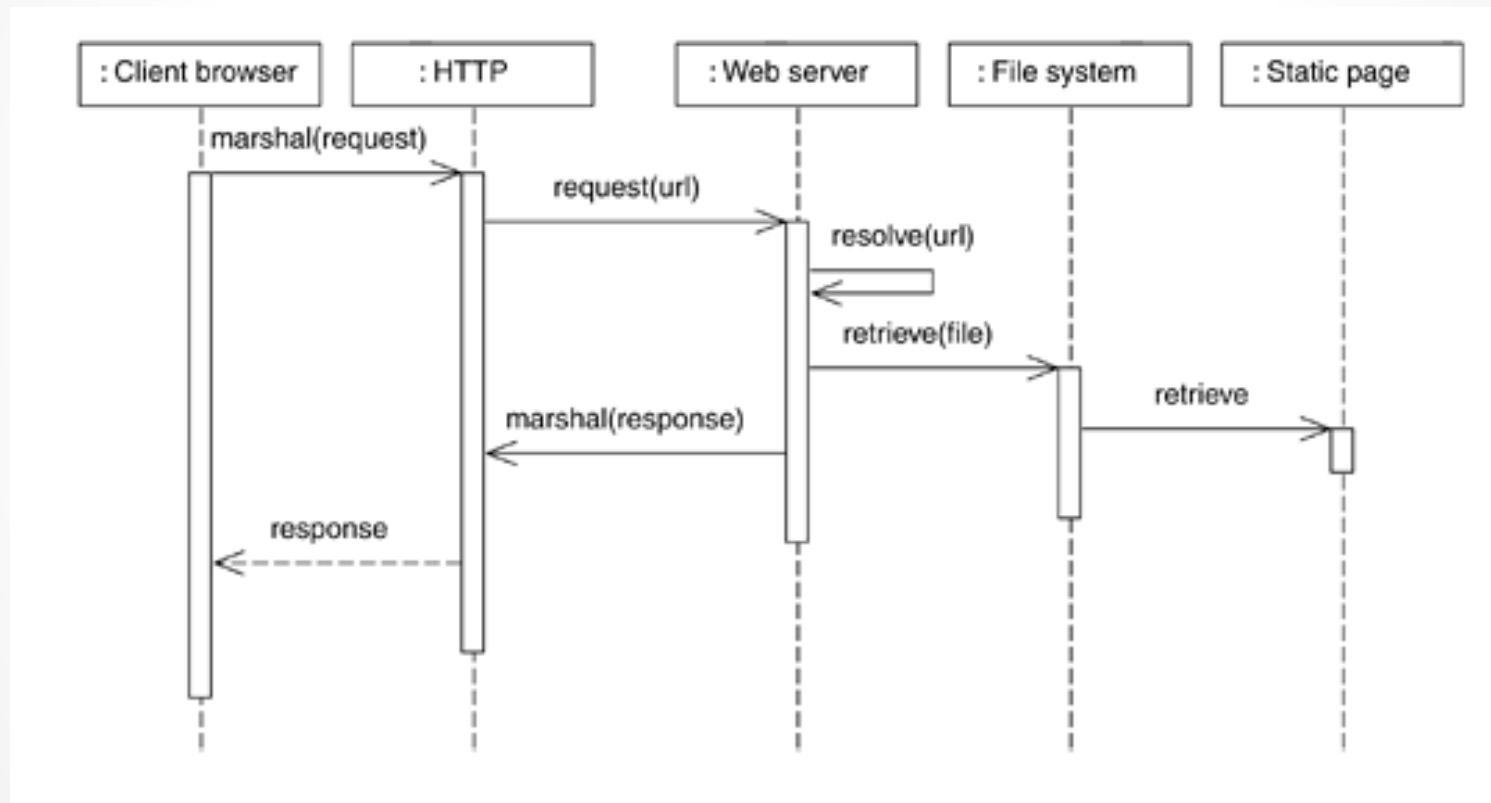
- **File system:**

- Untuk banyak aplikasi Web, sistem file adalah anggota kelas arsitektur, karena struktur direktori yang sering mencerminkan struktur URL logis.

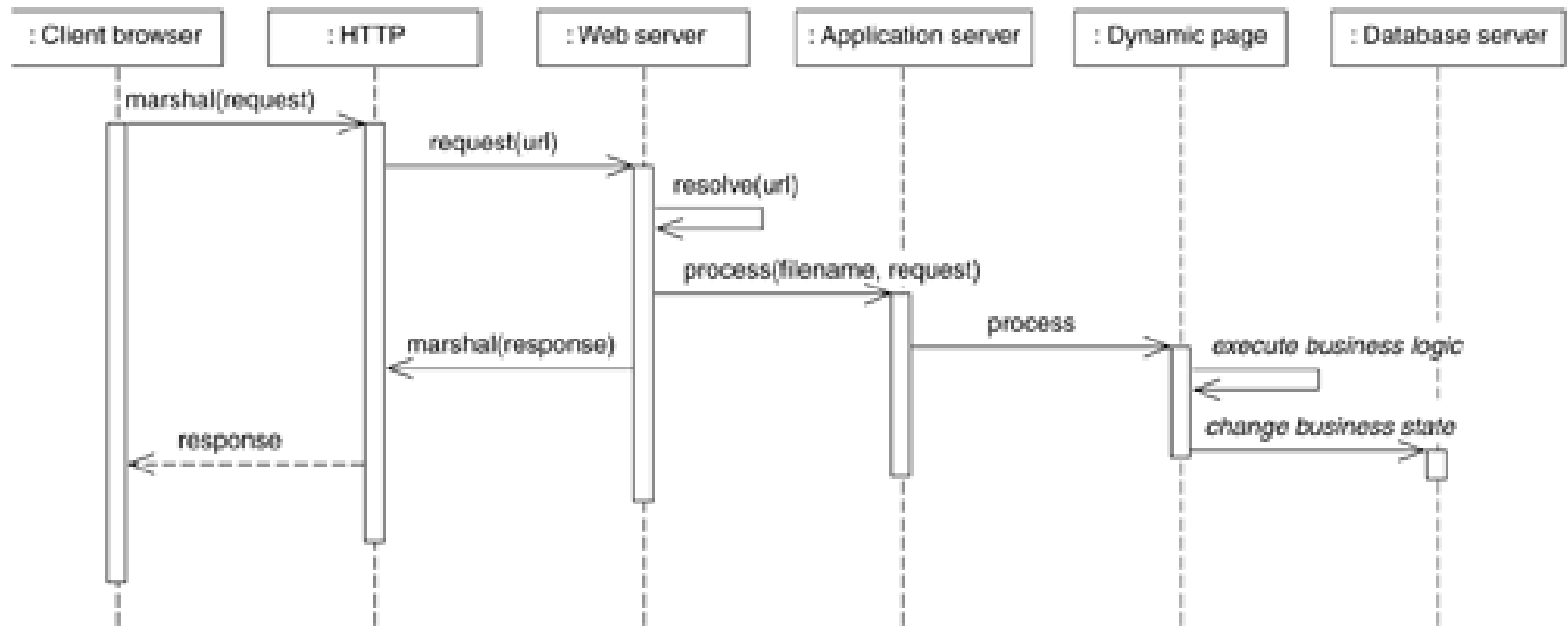
Principal participants in thin Web client collaboration



Static resource (page) request scenario



Dynamic resource (page) request scenario

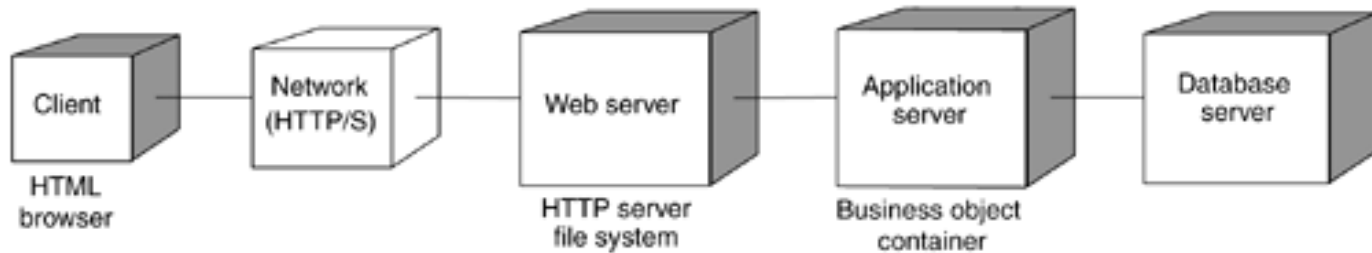


Thin Web Client

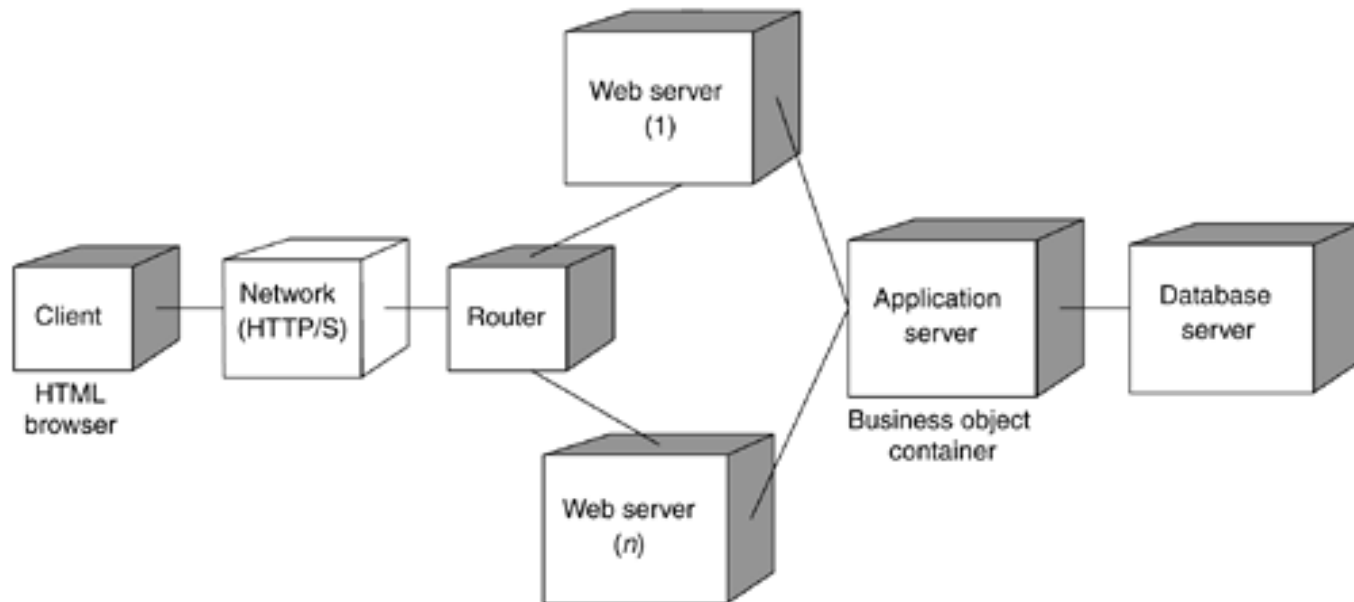
Sudut Pandang Realisasi

- Dapat sangat bervariasi, tergantung pada strategi skalabilitas.

Deployment of simple thin Web client application



Deployment of scalable thin Web client application



Thin Web Client

Sudut Pandang Pengujian

- **Fokus utama:** untuk memverifikasi bahwa sistem tersebut memenuhi kebutuhan para pemangku kepentingan, seperti yang ditentukan oleh persyaratan.
- Untuk aplikasi Web, banyak daerah yang membutuhkan pengujian mungkin tidak disebutkan dalam persyaratan.
 - Biasanya, daerah-daerah tersebut adalah hasil dari konteks sistem, seperti penggunaan di Internet dengan versi browser yang tidak diketahui.
- Untuk thin Web client architectures, area kunci yang perlu ditangani selama pengujian adalah
 - Semua versi browser yang didukung
 - Pengaturan firewall: penggunaan hanya port standar dan protokol
 - Jaringan latency
 - ISP and browser page caching
 - Cookie nature and use

Thick Web Client

- Memperpanjang Thick Web Client Pattern dengan client-side scripting seperti kontrol ActiveX dan applet Java
- Klien dapat menjalankan beberapa logika bisnis dari sistem dan dengan demikian menjadi lebih dari antarmuka pengguna umum terkait
- Paling sesuai untuk aplikasi Web di mana:
 - konfigurasi klien tertentu dan versi browser dapat diasumsikan,
 - user interface canggih yang diinginkan, dan/ atau
 - sejumlah logika bisnis dapat dijalankan pada klien.

Thick Web Client (2)

Sudut Pandang Persyaratan

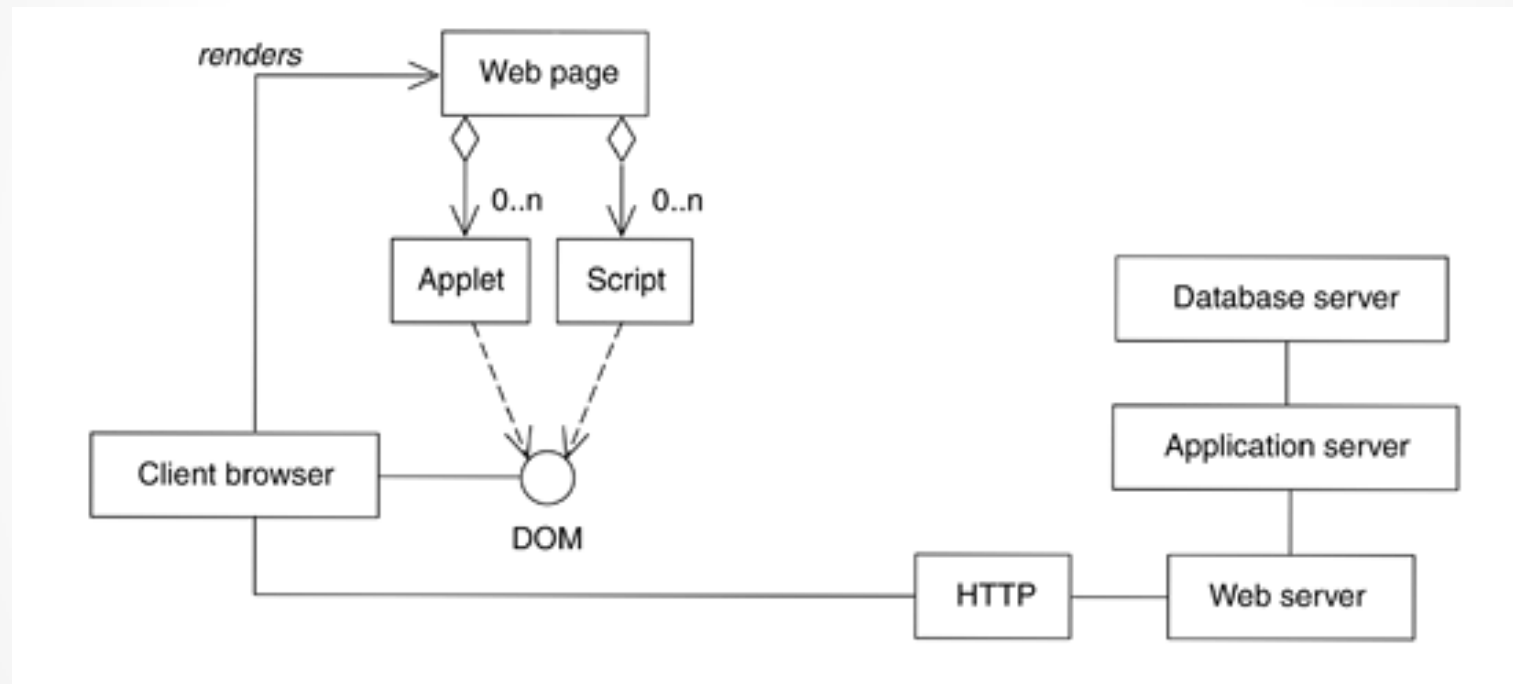
- Tambahan persyaratan baru: kemampuan untuk menjalankan beberapa logika sistem pada klien
- Contoh yang paling dasar: script atau applet yang memvalidasi bidang bentuk ini sebelum diserahkan ke server.
 - Bila mungkin, menghentikan bentuk-bentuk yang tidak valid dari yang disampaikan ke server dapat meningkatkan kinerja sistem.
 - Skrip tambahan dan kontrol pada klien juga dapat memberikan bantuan cerdas dalam mengisi formulir yang rumit atau dalam menjelajahi halaman sistem.
- Pengalaman antarmuka pengguna klien dapat ditingkatkan dengan aktivitas sisi klien
 - Kontrol antarmuka pengguna yang canggih, atau scripting, dapat membuat user interface yang lebih mudah untuk menggunakan atau lebih menarik
 - Interface yang menarik adalah syarat penting dalam ruang e-commerce yang ketat, di mana hanya satu klik

Thick Web Client (3)

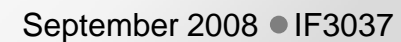
Sudut Pandang Perancangan

- Semua komunikasi antara klien dan server, seperti dalam pola Web thin client, dilakukan melalui HTTP atau HTTPS. Karena protokol ini adalah "connectionless," sebagian besar waktu, tidak ada hubungan terbuka antara klien dan server. Hanya selama permintaan halaman yang klien mengirim informasi dan permintaan ke server.
- Ini berarti bahwa klien-side scripting, kontrol ActiveX, dan Java applet terbatas berinteraksi dengan objek hanya pada klien.
- Menggunakan kemampuan browser tertentu, seperti Java applet atau kontrol ActiveX, untuk mengeksekusi logika bisnis pada klien.
- Komponen-komponen utama dalam arsitektur thick web client adalah sama seperti yang di web thin client, dengan penambahan script, applet, dan Document Object Model (DOM) antarmuka yang didukung oleh browser klien.

Principal participants in thick Web client architecture



● 45

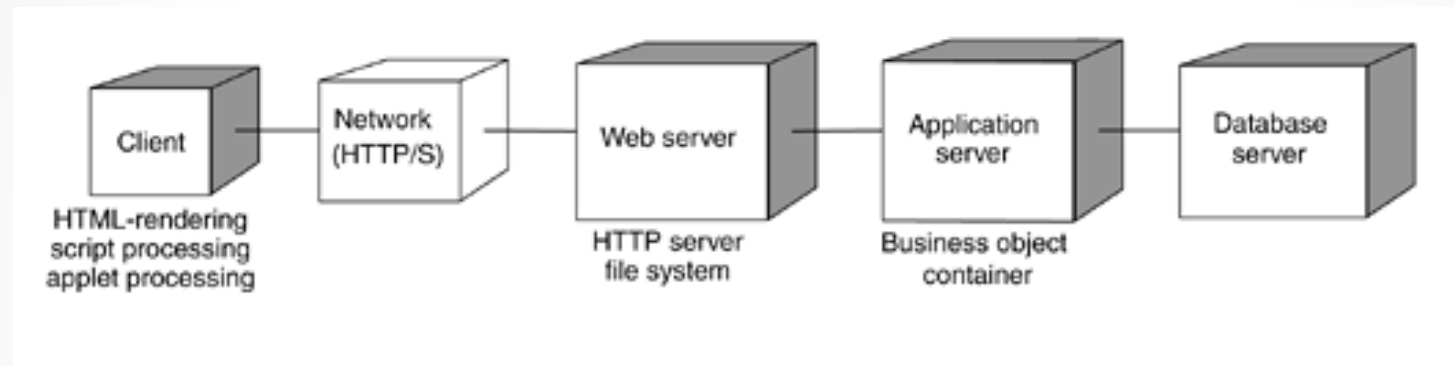


Thick Web Client (4)

Sudut Pandang Realisasi

- Hampir identik dengan yang ada pada thin Web client
- Architectural's node:
 - Client,
 - Network,
 - Web server,
 - Application server, and
 - Database server.
- Satu-satunya perbedaan: Node klien bertanggung jawab untuk script pengolahan dan mengeksekusi applet

Deployment of simple thick Web client application



Thick Web Client (5)

Sudut pandang Pengujian

- Pengujian dan jaminan kualitas perlu untuk mengatasi beberapa area tambahan:
 - Variasi dalam mesin vendor 'scripting
 - Akses untuk mendukung kelas Java atau libraries
 - Jaringan latency dan dampaknya pada pengolahan naskah multithreaded
 - Pemeriksaan yang cermat dari situasi kesalahan, atau hilangnya komponen applet
- Salah satu situasi yang pengembang sering mengabaikan perilaku sistem ketika sumber daya tertentu, seperti frame atau applet terpisah, tidak di-load dengan benar
 - Hal ini terjadi cukup sering dalam aplikasi yang beroperasi melalui Internet
 - Tim pengujian perlu memastikan bahwa sistem berperilaku anggun ketika applet tidak memuat dalam waktu

Web Delivery

- Web digunakan terutama sebagai mekanisme pengiriman untuk objek klien/ server sistem didistribusikan.
- Dari satu sudut pandang, jenis aplikasi adalah klien aplikasi/ server terdistribusi objek yang terjadi untuk menyertakan Web server dan browser klien sebagai elemen arsitektur yang signifikan.

Web Delivery (2)

Sudut Pandang Persyaratan

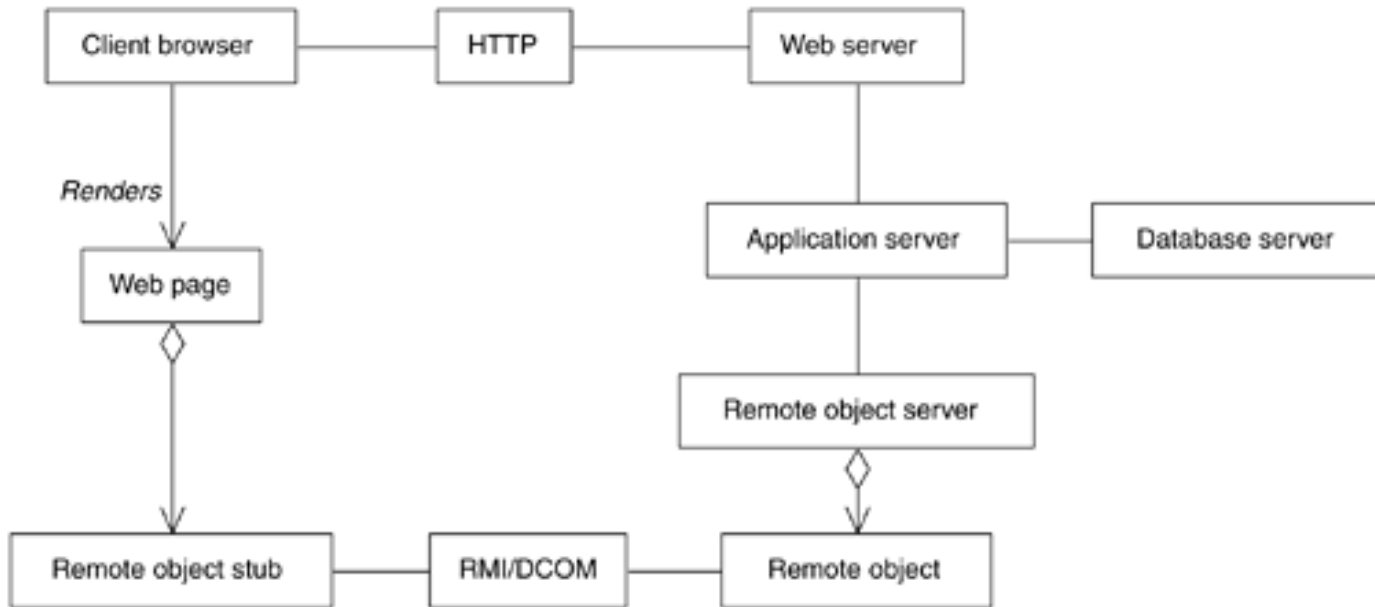
- HTTP bisa menjadi protokol yang sangat membatasi untuk aplikasi yang paling canggih. Overhead yang dibutuhkan oleh lalu lintas HTTP standar akan membuat mustahil bagi banyak peralatan-monitoring aplikasi real-time yang berfungsi di Internet.
- Kadang-kadang, good old-fashioned remote procedure calls or remote operation calls adalah solusi terbaik.
- Paling tepat ketika terdapat pengendalian klien dan jaringan konfigurasi.
- Kekuatan terbesar: kemampuannya untuk memanfaatkan obyek bisnis yang ada dalam konteks aplikasi Web.

Web Delivery (3)

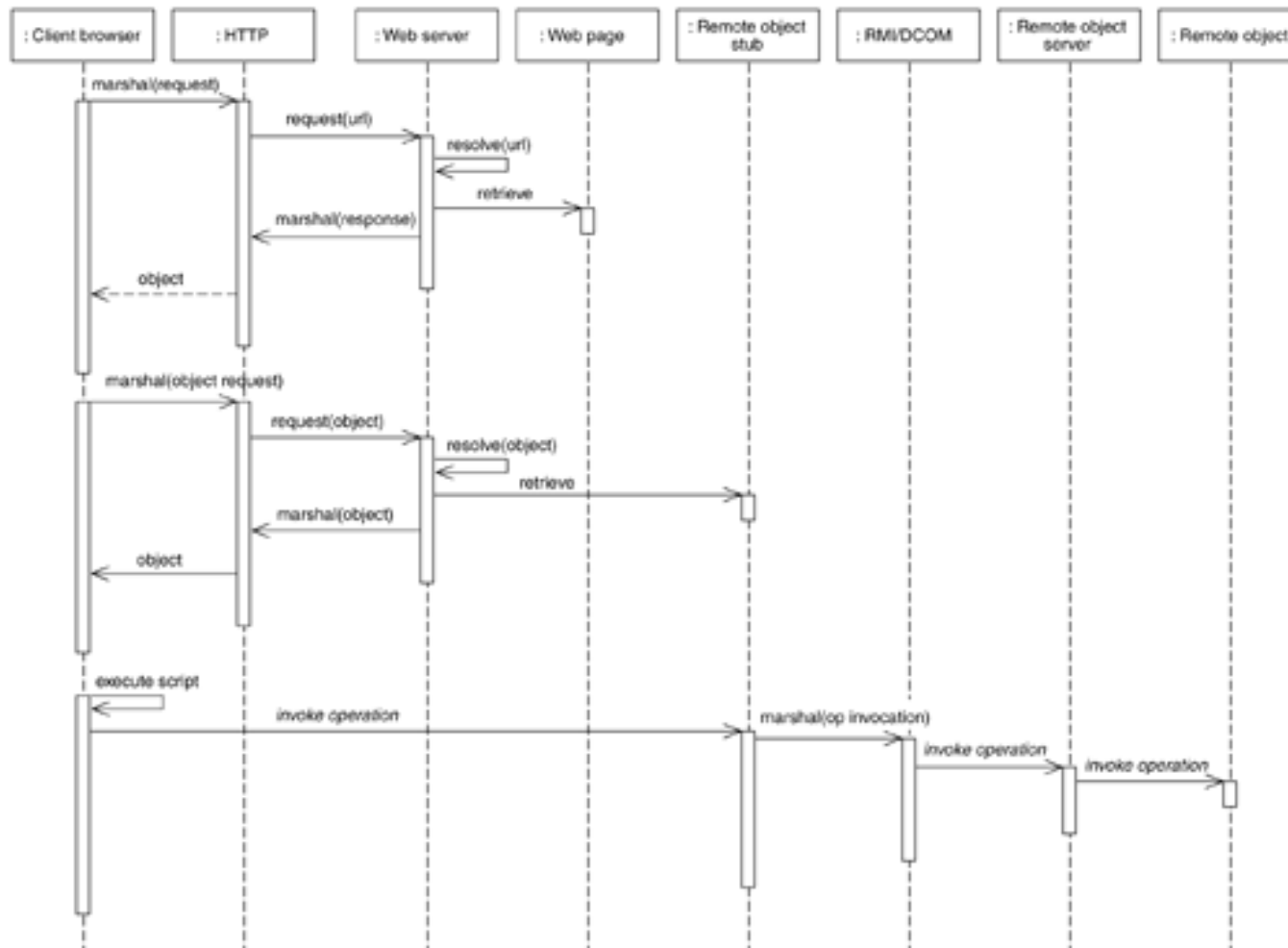
Sudut Pandang Perancangan

- Perbedaan yang paling signifikan antara pengiriman Web dan pola arsitektur aplikasi Web lainnya adalah metode komunikasi antara klien dan server
 - Dalam pola lain, mekanisme utama adalah HTTP, protokol connectionless yang sangat membatasi desainer ketika datang ke aktivitas interaktif antara pengguna dan server
- Unsur-unsur arsitektur signifikan dalam pola pengiriman Web mencakup semua yang ditentukan dalam dua pola sebelumnya, sebagai berikut:
 - **Remote object/remote object stub:** Sebuah benda konseptual yang dapat berinteraksi dengan, yang eksekusi berlangsung pada mesin remote.
 - **Remote object server:** Sebuah server khusus yang host objek remote.
 - **Remote object transfer protocol:** Protokol yang memungkinkan akses efisien untuk objek remote melalui jaringan standar
 - Java RMI
 - DCOM

Principal participants in the Web delivery architecture



Basic Web delivery remote object execution scenario

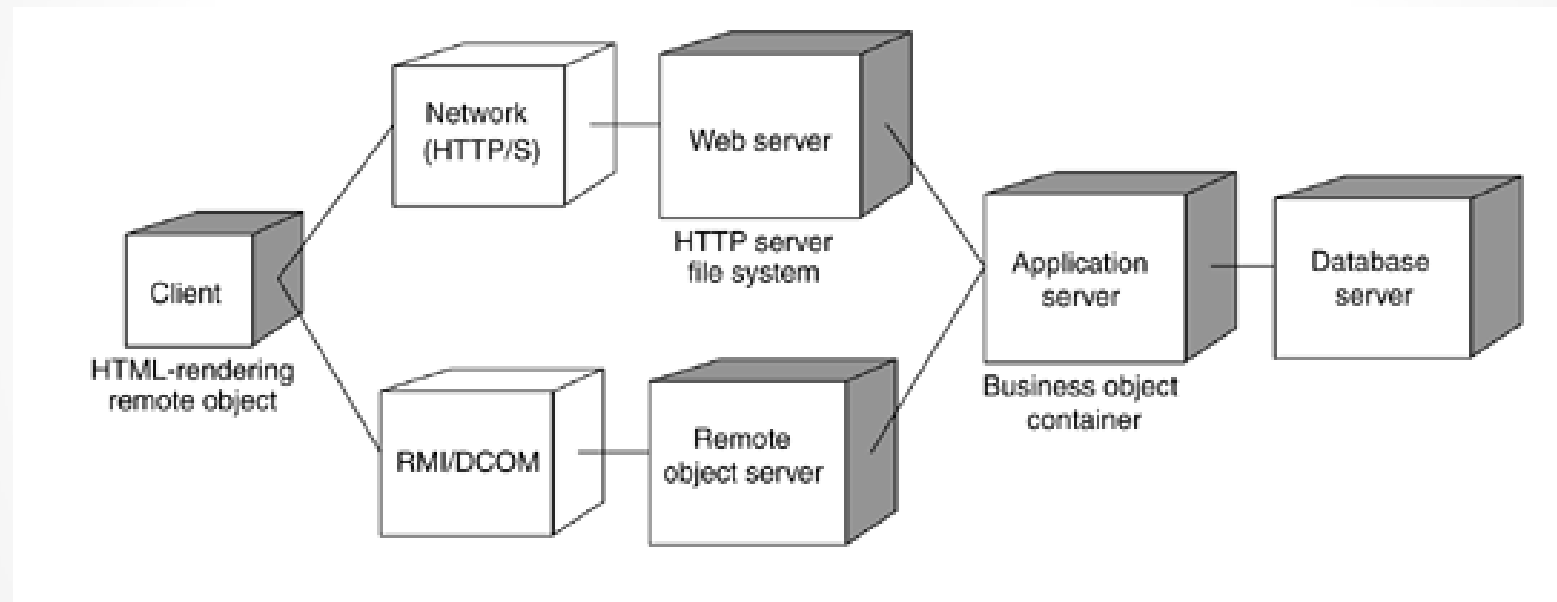


Web Delivery (4)

Sudut Pandang Realisasi

- Memperkenalkan node baru dan perangkatnya
 - Server remote object bertanggung jawab untuk membuat objek remote diakses oleh klien melalui jaringan
 - Protokol atau method-invocation marshaling mechanism adalah protokol RMI atau DCOM
- Perlu dicatat bahwa dalam beberapa situasi, server aplikasi dan server remote object mungkin satu dan sama dalam sudut pandang realisasi

Deployment of simple Web delivery application



Web Delivery (5)

Sudut Pandang

- Masalah pengujian yang relevan dengan arsitektur pengiriman Web mencakup pemantauan cermat infrastruktur yang dibutuhkan untuk melaksanakan panggilan jarak jauh
- Karena kebutuhan sering merupakan hasil dari kebutuhan untuk cepat, tahan lama, dan dapat diandalkan komunikasi langsung dengan server, pemeriksaan yang teliti terhadap latency jaringan dan toleransi kesalahan penting ketika menguji aplikasi ini
- Pengujian dan jaminan kualitas harus mengatasi beberapa area tambahan seperti:
 - Penggunaan port tambahan untuk remote object komunikasi melalui firewall
 - Kinerja melalui koneksi jaringan yang lambat
 - Efek pada underpowered mesin klien

TERIMA KASIH

Minggu Depan

Silahkan Lanjutkan Laporan Tugas Besar:

Gambarkan setiap Diagram berikut berdasarkan tugas besar masing-masing kelompok:

1. Use case Diagram
 - Tentukan table use case untuk setiap use case (ikuti format)
2. Class Diagram
3. Activity Diagram
 - Gambarkan Activity diagram untuk setiap use case (10 use case = 10 activity diagram)
4. Sequence Diagram
 - Gambarkan Sequence diagram untuk setiap use case (10 use case = 10 sequence diagram)

Gambarkan menggunakan Enterprise Architect

Print Laporan tanpa jilid

Presentasi Setiap Kelompok minggu depan

Format Tabel Use Case

Nama use case	Tambah Data Petugas	
Prioritas	Tinggi	
Aktor	Administrator	
Deskripsi	Use-case ini memungkinkan aktor untuk menambah petugas.	
Pre Kondisi	Aktor belum melihat halaman tambah data petugas.	
Sasaran	Use-case ini akan berfungsi bila aktor telah memberi perintah kepada system untuk menambah data petugas.	
Bidang khas suatu acara	Aktor Action	Respon Sistem
	Aktor memilih menu tambah data petugas.	Sistem menampilkan tambah data petugas.
Bidang Alternative	Bila aktor tidak memilih menu tambah data petugas, sistem tidak menampilkan halaman tambah data petugas.	
Kesimpulan	Use case ini selesai bila halaman tambah data petugas sudah muncul.	
Pos Kondisi	Aktor telah memilih menu tambah data petugas dan halaman tambah data petugas sudah ditampilkan.	
Aturan Bisnis	Yang berhak melihat tambah data petugas adalah Administrator.	
Spesifikasi dan Implementasi	Form-form yang ditampilkan harus user-friendly, mudah dan cepat untuk dimengerti dan digunakan.	